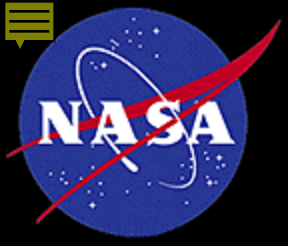
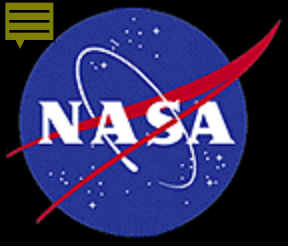


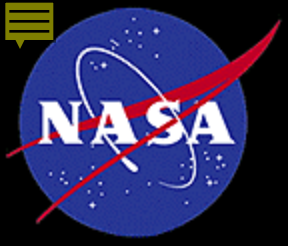
Using class, activity, and state charts to validate system requirements : *the basics*

John “Mac” Felsing



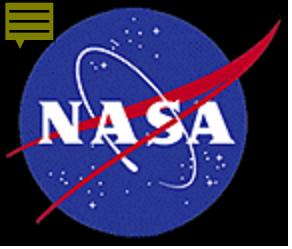
Objectives

- Show how multiple sources of information are linked and synthesized into an information repository (model).
- Demonstrate how the viewing the synthesized information in a graphical format provides a more powerful analysis perspective
- Show how the information helps to identify complex risks, issues, mismatches, missing information, and inconsistencies that can jeopardize project success.
- Lays the ground work for developing both static and dynamic analysis tools IV&V can use to produce products for our customers.



Why We Model

- Creating models helps us to understand the system under consideration.
- It helps us with the 2 C's (Complete and Consistent)
  - This is a byproduct of the modeling process itself
- It exposes a clearer view of the relationships between the many requirements, specifications, elements, systems, subsystems components, behaviors and interactions.
- It exposes disconnects, missing elements, conflicts, etc. very early on in the programs life cycle.
- The visual presentation format allows us to present different views and perspectives of complex systems of systems and gives us a better understanding of those systems and their interactions.

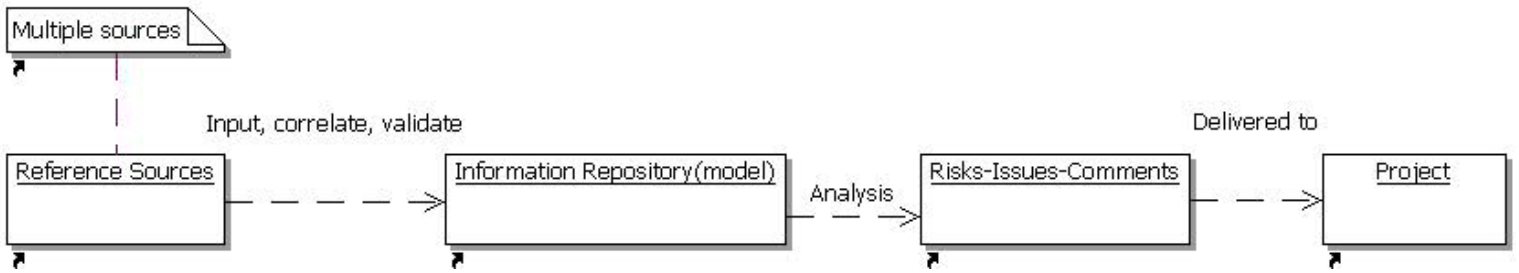


What modeling produces

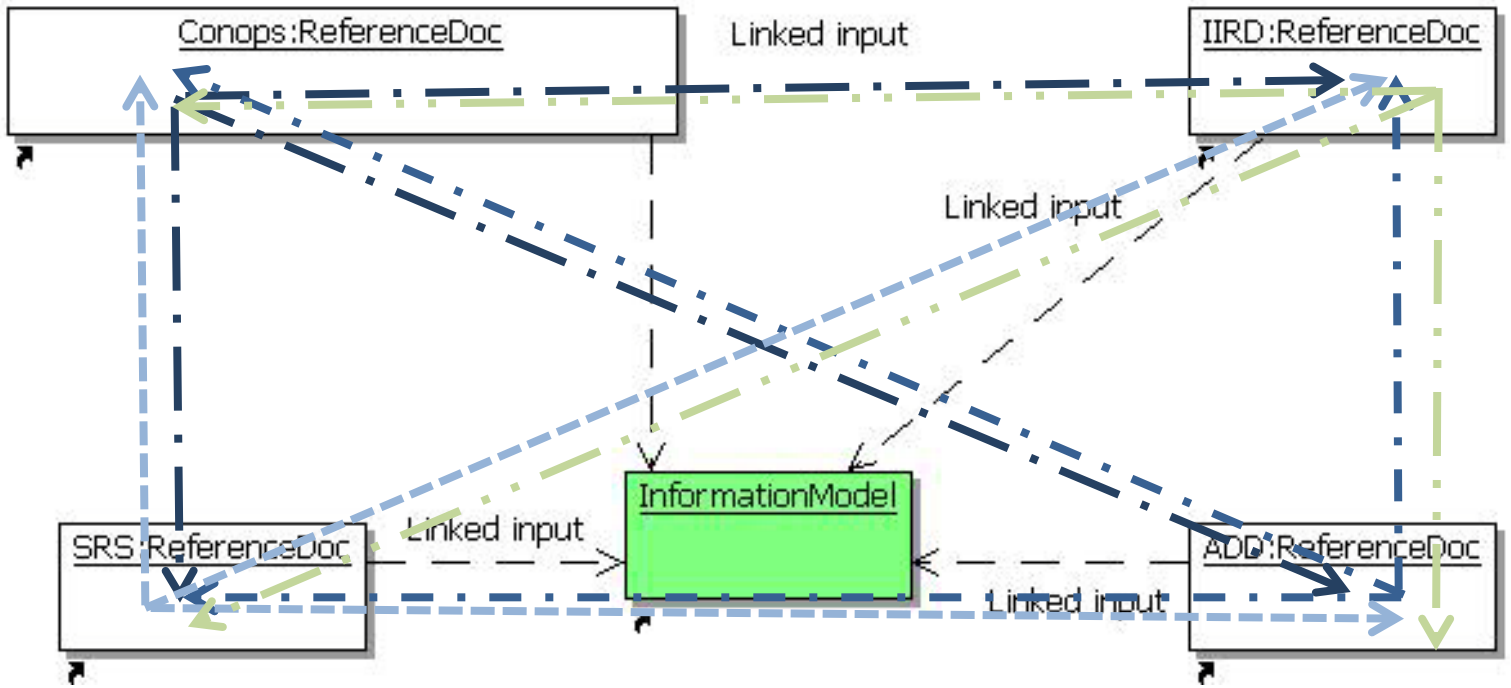
- A **logical information repository** that can be mined and analyzed to answer the 3 questions.
- At IV&V models are NOT our end product.
  - We generate questions which may turn into risks, issues and comments to be given back to the projects to identify and prevent possible catastrophic failures, Loss of Life, Loss of Mission, and Loss of Assets.
- The model is a **tool** to used to help us provide our IV&V products
- Uncover and understand additional scenarios
  - Quickly generate obvious scenarios and explore scenarios that may not be well understood at any one point during the project.
- Provides analysis evidence
- Create various reports about the content/consistency of the information available

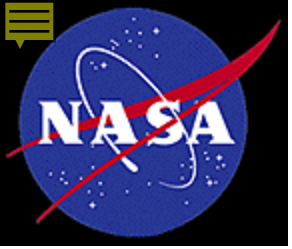
Information Flow

- Sources – (multiple)
- Flow into the Model (Information Repository)
- Output from the model
  - Risks, Issues, Comments
- Delivered to the project



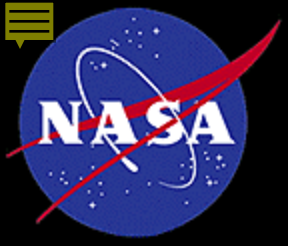
Multiple Sources





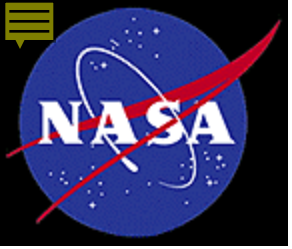
What information are we hunting?

- We look for information (requirements, operational behaviors, interactions, etc) that are:
  - Orphaned
  - Missing
  - Conflicting
  - Ambiguous
  - Incomplete
  - Out of place



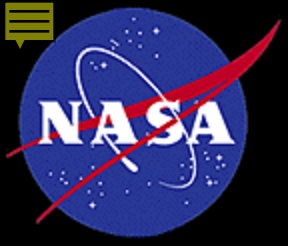
Setting up the information – the Basics

- Classes/Objects
- Activities/Behaviors
- States / Transitions



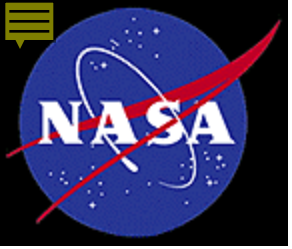
Classes / Objects

- **Class** – pattern or blueprint of a related set of real world objects.
  - Self aware - Attributes
  - Behaviors (Methods) - Links to other objects.
- **Objects**
  - An object is something that occurs in the real world.
  - The instantiation of a class.
  - A Software Object is something in a system that models a real world object. The object may be physical, or logical



Activities / Behaviors

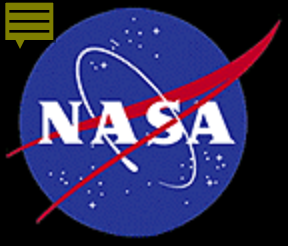
- Activities or Behaviors
  - A sequence of labeled events or actions
  - Include steps necessary to accomplish the behavior
  - Pre and Post Conditions
  - Guards and Constraints
  - Nominal and off Nominal flows
  - Data stores
  - Actors (Objects in the form of swim lanes)
  - Interactions between actors (interface)



States and Transitions

- State – value of the attributes of an instantiated class at a given point in time
- Transition – the change in values of the attributes of an object

- Identifies the known states of an object
- Identify the actions that cause transitions between known states



How we combine the information

- Transitions on a state diagram link to action steps in and activity diagram
- Partitions link to classes and objects
- Activities link to behaviors

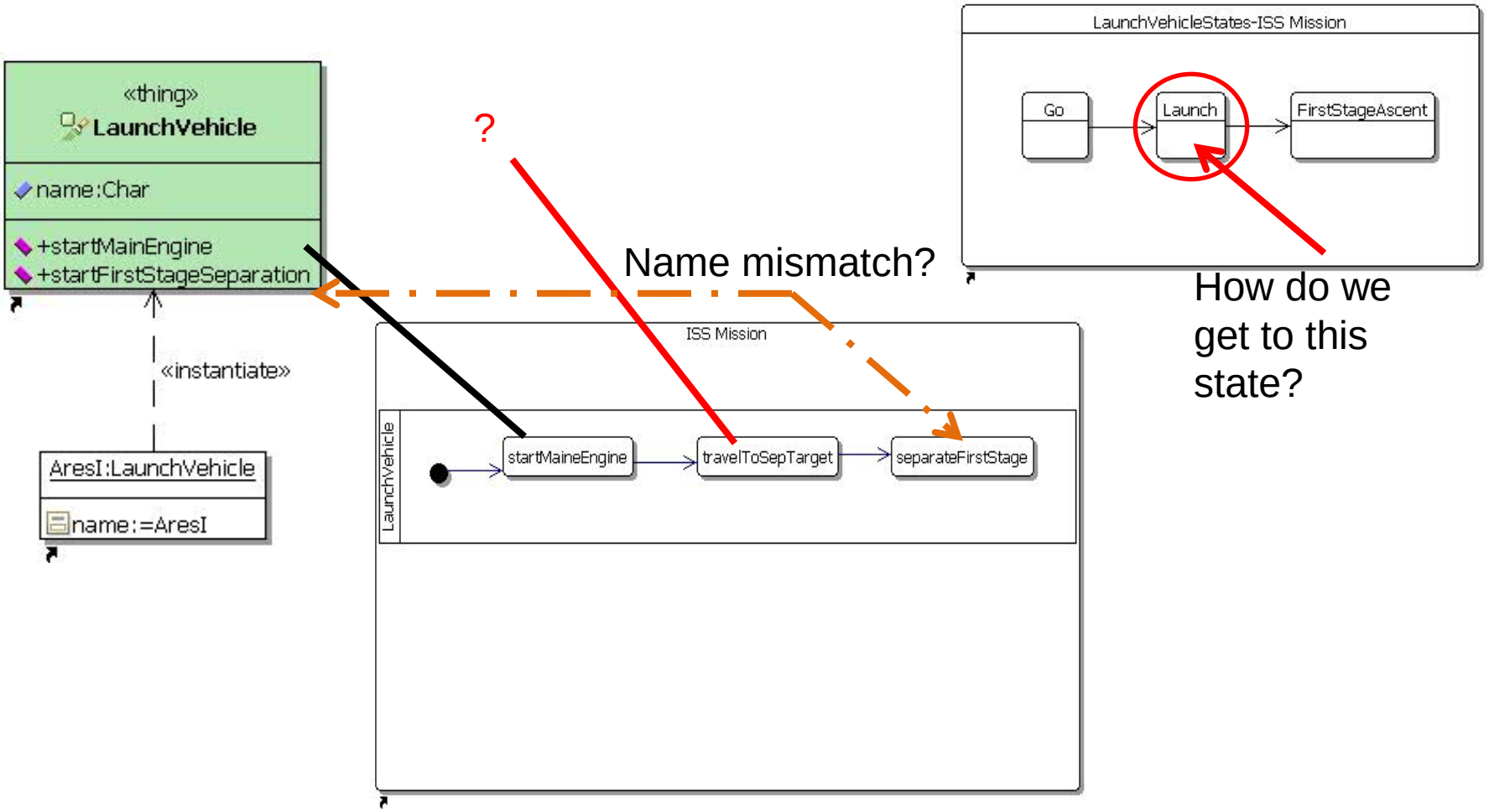
- They are all views of the same information

Linked Information

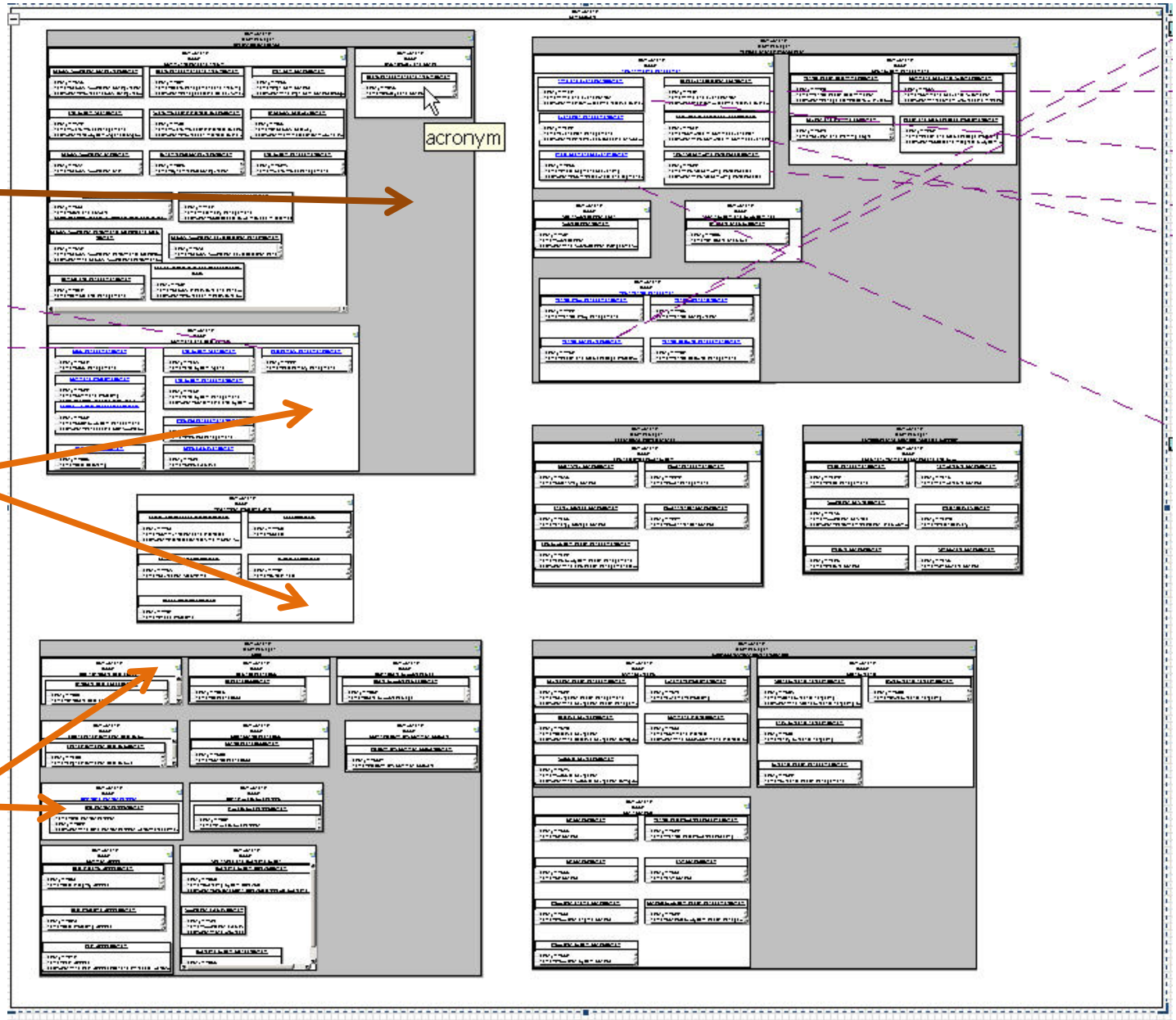
Class

Activity

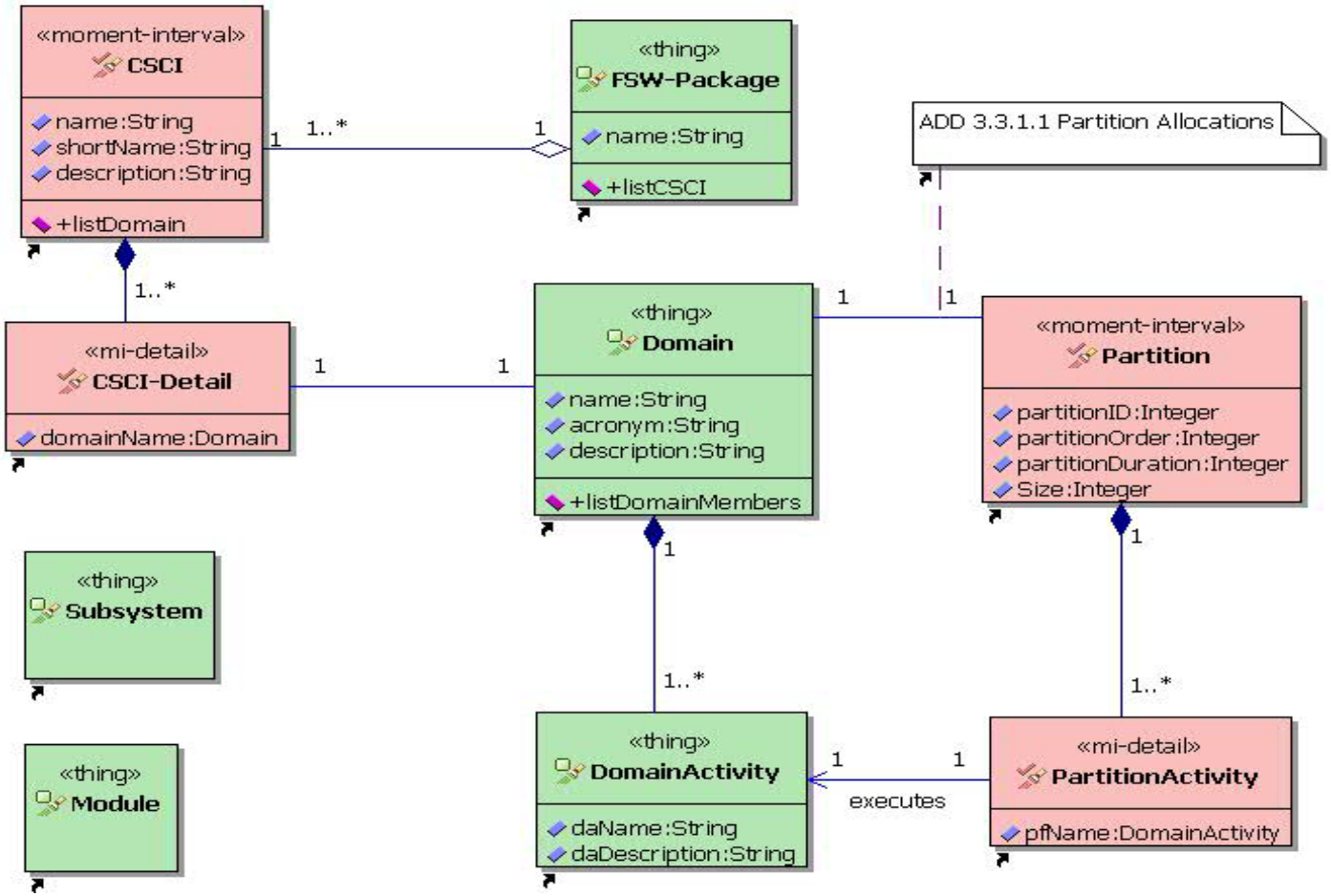
State



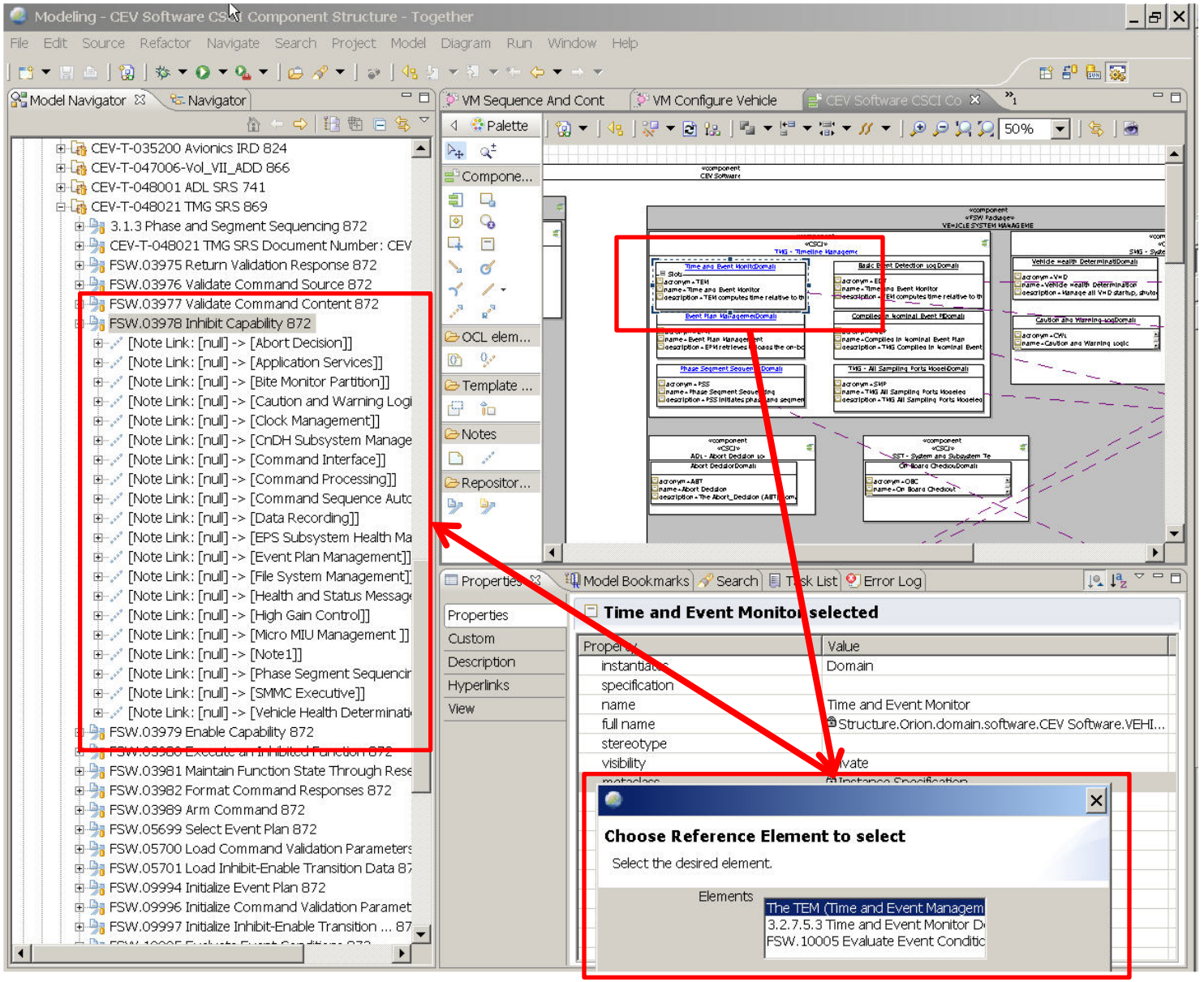
Orion CEV Software

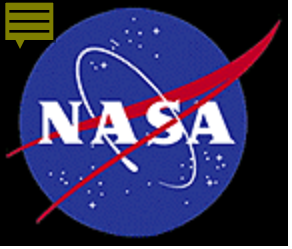


Orion Software Organization Structure



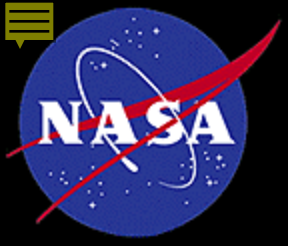
Tracing from Reference to Object





Ask questions, use model to answer

- Reports on issues
  - References
  - Observations
  - Orphaned elements
  - Missing elements
  - Conflicting elements



Summary

- Model is an **Information Repository**
  - Allows a more omniscient view of system
  - Find complex inconsistencies and issues among multiple sources at one time
  - Identify areas of ambiguity quickly
  - Justify Issues and discrepancies, their importance and criticality

Next Steps

- Animation/Dynamic Behavior
  - Validate that a given activity and states support the specified behavior through animation/simulation
- More transforms to find more inconsistencies and issues.
  - Generate classes, interfaces, and component diagrams from activity and state diagrams.
- Automate generation and identification of issues through audits and metrics
  - Use model audits and metrics to quickly identify discrepancies and missing or conflicting information at the diagram level.
- Impact analysis
  - Identify the impact of changing requirements, design and architecture using the information in the model to quickly identify areas that are affected by changes in reference documentation and requirements
- Other
  - Create report templates and queries to quickly extract information for IV&V products, such as issues, observations, missing or conflicting requirements, analysis views, etc.